



WAVELET REGULARIZATION OF A FOURIER-GALERKIN METHOD FOR SOLVING THE 2D INCOMPRESSIBLE EULER EQUATIONS

Romain Nguyen van Yen, Marie Farge, Kai Schneider

► To cite this version:

Romain Nguyen van Yen, Marie Farge, Kai Schneider. WAVELET REGULARIZATION OF A FOURIER-GALERKIN METHOD FOR SOLVING THE 2D INCOMPRESSIBLE EULER EQUATIONS. ESAIM: Proceedings, 2009, 29, pp.89-107. 10.1051/proc/2009057 . hal-00601723

HAL Id: hal-00601723

<https://hal.science/hal-00601723>

Submitted on 20 Jun 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

WAVELET REGULARIZATION OF A FOURIER-GALERKIN METHOD FOR SOLVING THE 2D INCOMPRESSIBLE EULER EQUATIONS^{*,**}

R. NGUYEN VAN YEN¹, M. FARGE¹ AND K. SCHNEIDER²

Résumé. Nous appliquons une méthode Fourier-Galerkin pour résoudre les équations d'Euler 2D incompressibles, et étudions plusieurs façons de régulariser la solution par filtrage en ondelettes à chaque pas de temps. Nous considérons des ondelettes orthogonales à valeurs réelles, ainsi que des ondelettes à valeurs complexes. utilisées pour effectuer soit un filtrage linéaire soit un filtrage non linéaire. Les résultats sont comparés à ceux obtenus par des méthodes classiques de régularisation visqueuse ou hypervisqueuse. Ils montrent que la régularisation par ondelettes complexes se comporte aussi bien en termes de taux de convergence L^2 vers la solution de référence. Le taux de compression pour la turbulence 2D homogène isotrope obtenu avec cette méthode est de l'ordre de 3, de sorte que l'espace mémoire et le temps de calcul nécessaires seraient inférieurs pour une simulation adaptative en ondelettes. Nos résultats suggèrent également que la convergence vers la solution de référence a lieu même sans aucune régularisation, contrairement à ce qui se produit pour l'équation de Burgers 1D.

Abstract. We employ a Fourier-Galerkin method to solve the 2D incompressible Euler equations, and study several ways to regularize the solution by wavelet filtering at each timestep. Real-valued orthogonal wavelets and complex-valued wavelets are considered, combined with either linear or non-linear filtering. The results are compared with those obtained via classical viscous and hyperviscous regularization methods. Wavelet regularization using complex-valued wavelets performs as well in terms of L^2 convergence rate to the reference solution. The compression rate for homogeneous 2D turbulence is around 3 for this method, suggesting that memory and CPU time could be reduced in an adaptive wavelet computation. Our results also suggest L^2 convergence to the reference solution without any regularization, in contrast to what is obtained for the 1D Burgers equation.

INTRODUCTION

When they are dominated by nonlinear effects due to inertia, flows in Newtonian incompressible fluids enter the turbulent regime, in which their motions are very disordered and involve a wide range of scales. Nevertheless, they tend to possess certain robust phenomenological properties, e.g. the average energy dissipation rate, that have lead to their study in a specific framework known as the statistical theory of fully developed homogeneous isotropic turbulence [18]. This theory is not directly derived from the basic equations of fluid mechanics but

* The authors acknowledge financial support from the Euratom-CEA association under contract V.3258.006. This work, supported by the European Communities under the contract of Association between EURATOM, CEA and the French Research Federation for Fusion Studies, was carried out within the framework of the European Fusion Development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

** Computations have been carried out in part at the IDRIS national computing center (CNRS, Orsay, France).

¹ Laboratoire de Météorologie Dynamique-IPSL-CNRS, École Normale Supérieure, Paris, France; e-mail : rnguyen@lmd.ens.fr

² Centre de Mathématiques et d'Informatique, Université d'Aix-Marseille, France

requires additional statistical hypotheses that are difficult to check experimentally. Moreover, it doesn't take into account self-organization of the flow into coherent structures. On the other hand, current numerical methods tend to reproduce observed properties of turbulent flows. They offer rich opportunities for proposing new theoretical approaches and confronting them quantitatively.

Despite the chaotic character of turbulent flows, it is commonly accepted that, if the density is constant, their velocity field $\mathbf{u}(\mathbf{x}, t)$ is well described by solutions of the Navier-Stokes equations:

$$\begin{cases} \partial_t \mathbf{u}_\nu + (\mathbf{u}_\nu \cdot \nabla) \mathbf{u}_\nu &= -\nabla p_\nu + \nu \nabla^2 \mathbf{u}_\nu, & \mathbf{x} \in \mathbb{T}^d, \quad t \in]0, +\infty[\\ \nabla \cdot \mathbf{u}_\nu &= 0 \\ \mathbf{u}_\nu(\cdot, 0) &= \mathbf{u}_0 \end{cases} \quad (1)$$

where ν is the kinematic viscosity of the fluid, t is time, and \mathbf{u}_0 is an initial flow on the d -dimensional torus $\mathbb{T}^d = (\frac{\mathbb{R}}{2\pi\mathbb{Z}})^d$. Here, the pressure p_ν is determined by \mathbf{u}_ν through the divergence-free condition and cannot be chosen independently. The equations have been written in dimensionless units, so that the Reynolds number can be simply defined by $\text{Re} = \nu^{-1}$. Properly describing the behavior of solutions to (1) when $\text{Re} \gg 1$ remains a central problem for the understanding fully developed turbulence. A key mathematical difficulty is that the limit $\nu \rightarrow 0$, i.e. $\text{Re} \rightarrow \infty$, is singular as the order of the equation changes. Indeed, many wind tunnel experiments [4, 42], as well as numerical experiments [25], suggest that the limit \mathbf{u} of \mathbf{u}_ν when $\text{Re} \rightarrow \infty$ does not satisfy Euler's equations (2),

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla p \quad (2)$$

or at least not in the strong sense. This idea was elaborated upon by many authors following the seminal work of Onsager, about which a nice historical review has been recently published [13].

When trying to approximate numerically turbulent flows, the traditional approach is to solve the Navier-Stokes equations and make the viscosity as small as is allowed by numerical discretization, which is ultimately limited by the available computational power (see e.g. [25] for state of the art results obtained using 4096^3 degrees of freedom). But if the dynamics in the inviscid limit, or at least some of its important features, become strictly independent of Re , it is much more desirable to solve the Euler equations instead of the Navier-Stokes equations. A problem of parabolic type (eq. 1) is thereby replaced by a problem of hyperbolic type (eq. 2), as the higher order derivative disappears. In some cases, finite time singularities may then occur, and uniqueness of the solution is lost [31]. This happens for example for the Burgers equation, or for the compressible Euler equation, and is conjectured to happen for the 3D incompressible Euler equations. Admittedly, the 2D Euler equations, that we are going to consider below, do not give rise to finite time singularities. Their solution remains smooth for all time, since we are considering only domains without boundaries, and smooth initial data [27, 47]. However, the vorticity gradients grow extremely fast in time. This phenomenon, which has been described as "slow collapse" [48], is almost as bad as finite time singularity as far as numerical simulation is concerned.

Due to the roughness of the exact solutions to these problems, numerical methods for solving hyperbolic partial differential equations must be accompanied by proper regularization mechanisms. If the solution is not unique, the responsibility even falls upon the numerical scheme to select the weak solution that it approximates. Physical considerations must hence be taken into account when designing the scheme so that it yields the proper solution. Existing methods include upwind characteristic [38] or total variation diminishing [22] schemes, shock limiters [43], spectral vanishing viscosity [20, 44], hyperdissipation [3], and more recently inviscid regularization [2, 23, 29]. The latter stands aside from the rest since it was introduced mainly as a tool for mathematical proofs. One may also refer to [39] for a comparison between several schemes.

In this paper, we apply a regularization method based on wavelet filtering to solve the 2D Euler equations written in vorticity-velocity formulation:

$$\begin{cases} \partial_t \omega + \mathbf{u} \cdot \nabla \omega &= 0, \quad t \in]0, +\infty[, & \mathbf{x} \in \mathbb{T}^2 \\ \omega &= \nabla \times \mathbf{u} \\ \nabla \cdot \mathbf{u} &= 0 \\ \omega(\cdot, 0) &= \omega_0 \end{cases} \quad (3)$$

where ω denotes the vorticity component perpendicular to the plane. We compare the results with those obtained when the first equation is replaced by

$$\partial_t \omega_{\nu,\alpha} + (\mathbf{u}_{\nu,\alpha} \cdot \nabla) \omega_{\nu,\alpha} = -\nu_\alpha (-\Delta)^\alpha \omega_{\nu,\alpha}, \quad \alpha \geq 1 \quad (4)$$

which corresponds to the 2D hyperdissipative Navier-Stokes equations with dissipativity α if $\alpha > 1$, or to the classical 2D Navier-Stokes equations if $\alpha = 1$, in which case we drop the index α .

Note that the physical relevance of the inviscid limit in 2D is questionable, since virtually all known flows become three dimensional at high enough Reynolds number. Nevertheless, the 2D case is chosen as a toy model to allow us access to high Reynolds number flows at an affordable computational cost. Another unphysical aspect of our study resides in the boundary conditions. Physically realistic flows are in contact with solid walls which need to be taken into account by introducing no-slip boundary conditions in the modeling. These boundary conditions cannot be satisfied in general by solutions of the Euler equations. Taking them into account properly in the inviscid limit hence introduces new difficulties that we reserve for future work. In the meantime, we use periodic boundary conditions.

The main goal of this paper is to extend to the 2D Euler equations the study that we did in a previous work [35] concerning regularization of the 1D Burgers equation:

$$\begin{cases} \partial_t u + u \partial_x u &= 0, & t \in]0, +\infty[, & x \in [-1, 1], \\ u(\cdot, 0) &= u_0. \end{cases} \quad (5)$$

In the first part, we explain our numerical method, which consists in classical discretization schemes and the original part of our work, namely, the wavelet filters. In the second part, we present some new numerical results obtained for the 1D Burgers equation, and we then study in detail the regularization method in the case of the 2D Euler equations. Finally, we draw some conclusions relative to those two sets of results and outline perspectives for extending the study to 3D flows.

1. NUMERICAL METHOD

1.1. Discretization

In [35], the 1D Burgers equation was discretized by means of a Fourier collocation method, with pseudo-spectral evaluation of the nonlinear term [7]. In such schemes, the velocity is followed via its Fourier coefficients, which satisfy a system of nonlinear ordinary differential equations. To efficiently compute the convolution product that appears due to the quadratic nonlinear term, the solution is reconstructed on an evenly spaced grid, hence the term “pseudo-spectral”. Conservation of energy was enforced thanks to a skew-symmetric formulation of the nonlinear term. A fourth order Runge-Kutta time discretization and a small timestep allowed us to observe numerical dissipation of energy remaining below 10^{-4} relative to the initial energy, even for zero viscosity.

Here, we use a slightly different approach which applies both to the 1D Burgers and 2D Euler equations. A Fourier expansion with pseudo-spectral evaluation of the nonlinear term is still utilized, but the product in physical space is now fully dealiased. In other words, the Fourier modes retained in the expansion of the solution are such that $|\mathbf{k}| \leq k_C$, where k_C is the desired cut-off wavenumber, but the grid has $N = 3k_C$ points in each direction, versus $N = 2k_C$ for a non-dealiased, critically sampled product. This dealiasing makes the pseudo-spectral scheme equivalent to a Fourier-Galerkin scheme up to round-off errors [7], and thus conservative. For time discretization we stick to Runge-Kutta schemes, of order 4 for the 1D Burgers equation and of order 3 with a low storage formulation [37, page 20] for the 2D Euler equations.

All the required Fourier transforms are computed thanks to the FFTW library [17]. This allows us to exploit at the same time OpenMP and the Message Passing Interface (MPI) for parallelization. To reach resolutions up to 8192×8192 , 8 cores were utilized on a shared memory cluster node. For some of the simulations, we used 64 cores and 32 MPI processes on an IBM Regatta Power6 machine at the IDRIS-CNRS french computing center.

1.2. Regularization

We call “filtering scheme” any operator that can be applied to the discrete numerical solution at the end of each timestep, before advancing to the next timestep. In this subsection, we attempt to describe the filtering algorithms for a generic solution f , knowing that they will later be applied either to $f = u$ for the 1D Burgers case or to $f = \omega$ for the 2D Euler case. Since the filters that we are interested in are applied in the wavelet domain, we start by computing the transform coefficients of the solution with respect to the space variable (see the Appendix for technical details). Given orthogonal wavelets $(\psi_\lambda)_{\lambda \in \Lambda}$ and the associated scaling function φ at the largest scale, the solution is thus expanded into (see e.g. [33]):

$$f = \langle f | \varphi \rangle \varphi + \sum_{\lambda \in \Lambda} \langle u | \psi_\lambda \rangle \psi_\lambda, \quad (6)$$

where λ is a multi-index giving the scale, position, and, in 2D, direction of each wavelet. Later we denote by \tilde{f}_λ the wavelet coefficients $\langle u | \psi_\lambda \rangle \psi_\lambda$ of f . The inner product is defined in 1D by $\langle a | b \rangle = \int_{-1}^1 a(x) \cdot b^*(x) dx$, where $*$ stands for complex conjugation, and in 2D by $\langle a | b \rangle = \int_{\mathbb{T}^2} a(\mathbf{x}) \cdot b^*(\mathbf{x}) d\mathbf{x}$.

We consider two wavelet families. The first one, well known to numerical analysts, is the Coiflet orthonormal family with supports of length 12 [11], proposed in [14] to represent turbulent flows. The second one is the Qshift-B family associated with the dual-tree complex wavelet transform (DTCWT) [30], that we propose to dub Kingslet. Although it was primarily designed for image processing, the latter also offers many attractive features from a computational point of view. Its main characteristics can be summarized as follows:

- (1) the wavelets and scaling functions are complex-valued,
- (2) the family is not orthogonal, but it is a tight frame, which is a kind of overcomplete basis, having in d dimensions 2^d times more elements than an orthonormal basis at the same resolution,
- (3) near translation invariance is built into the transform, despite its dyadic structure,
- (4) in two and more spatial dimensions, the wavelets have much better directional selectivity than orthogonal wavelets.

Since the Kingslets constitute a tight frame, the reconstruction formula (6) is still valid [30], as long as the right hand side is replaced by its real part, f being real-valued.

Since this is the key point, let us explicit what we mean by translation invariance. Most multiscale approximation schemes involve the design of some sort of recursive grid or subdivision pattern. Normally these recursive structures depend on the choice of the origin and therefore are not invariant by translation. This is in particular the case for orthogonal wavelets, which are arranged as a dyadic tree. In most cases this tree has nothing to do with the phenomenon or signal one wants to study, but since it is built into the transform it tends to introduce some artifacts as soon as the coefficients are processed in some way. One solution consists in averaging the results over all possible shifts of the tree with respect to the signal. It is known as the undecimated wavelet transform [10]. Kingslets offer a more elegant solution to the same problem, namely, to average only a fixed number of trees (2^d in d dimensions), but under the condition that the wavelets associated to those trees are approximate Hilbert transform pairs. When this is done, the projectors on each approximation subspace almost commute with all possible signal shifts.

Once the wavelet coefficients of the solution have been obtained, we consider two kinds of filters: a linear one, and a nonlinear one. The linear filter consists in discarding only the wavelet coefficients at the finest scale. For the Kingslets case, this linear filter is equivalent to a simple convolution of the solution with the scaling function at the finest scale. It is nothing more than a lowpass filter, but with a special transfer function that happens to be the Fourier transform of a scaling function. This property does not hold for orthogonal wavelets, because the downsampling operation applied to the scaling function coefficients at the finest scale introduces aliasing effects [30].

For the nonlinear filter, before discarding the wavelet coefficients at the finest scale, we apply a threshold to all wavelet coefficients: those whose moduli are below a certain value Θ are set to zero. The threshold Θ depends on the function itself and is computed by means of an iterative method [1]. The resulting value is characterized

Type	α	ν_α	Wavelet	Filter
Viscous (i)	1	> 0	none	none
Hyperviscous (ii)	4	> 0	none	none
Real linear (iii)	-	$= 0$	Coiflet	linear
Complex linear (iv)	-	$= 0$	Kingslet	linear
Real CVS (v)	-	$= 0$	Coiflet	nonlinear
Complex CVS (vi)	-	$= 0$	Kingslet	nonlinear

TABLE 1. Summary of all employed regularization methods.

by the following implicit relationship:

$$\Theta^2 = \frac{5}{N_I(\Theta)} \sum_{\lambda \in \Lambda} |\tilde{f}_\lambda|^2 H(\Theta - |\tilde{f}_\lambda|), \quad (7)$$

where H is the Heaviside step function and $N_I(\Theta) = \sum_{\lambda \in \Lambda} H(\Theta - |\tilde{f}_\lambda|)$ is the number of wavelet coefficients below the threshold. Formula (7) simply means that the threshold equals 5 times the standard deviation of the coefficients below it. Note that there is no adjustable parameter. The nonlinear filter will be referred to as CVS filter, where CVS stands for Coherent Vorticity Simulation [16]. After applying either filter, the solution is reconstructed by inverse wavelet transform, and its Fourier coefficients by fast Fourier transform (FFT), so that the simulation can proceed with the next timestep.

The viscous or hyperviscous term, when present, is included in the computation without loss of accuracy thanks to an integrating factor method [46, page 111]. The integrating factor is simply $e^{\nu k^2 t}$, with k the Fourier wavenumber modulus.

The six different regularization methods that we are going to compare are:

- (i) Viscous: $\alpha = 1$ and $\nu_1 > 0$ is chosen high enough so that all scales of motion are resolved without applying any filter,
- (ii) Hyperviscous: $\alpha = 4$, and $\nu_4 > 0$ is defined by $\nu_4 = \nu_1 k_C^{-6}$,
- (iii) Real linear: $\nu = 0$, and we apply linear wavelet filtering of the finest scale using Coiflets,
- (iv) Complex linear: $\nu = 0$, and we apply linear wavelet filtering of the finest scale using Kingslets,
- (v) Real CVS: $\nu = 0$, and we apply nonlinear CVS filtering using Coiflets,
- (vi) Complex CVS: $\nu = 0$, and we apply nonlinear CVS filtering using Kingslets.

A summary is given in table 1. We shall also compare the regularized solutions to the one obtained without any regularization mechanism, that is, by cautiously applying our numerical scheme to the inviscid equation. The solution does not blow up even in this case since the numerical scheme is conservative. However, we are not claiming that it is the genuine solution of the – non-truncated – 2D Euler equations, and in fact it cannot be, due to the abrupt cut-off in Fourier space.

1.3. Initial conditions

For the 1D Burgers equation, we choose the initial condition $u_0(x) = -\sin(\pi x)$, which induces the formation at $t = \frac{1}{\pi}$ of a single shock located in the middle $x = 0$ of the interval $[-1, 1]$. For the viscous simulations **(i)**, we choose $\nu_1 = 2N^{-1}$. A classical estimate says that the shock width is proportional to ν_1 , hence ν_1 has to be larger than a constant times N^{-1} for the simulation to be properly resolved. The constant is adjusted by trial and errors. The value of ν_4 for all the hyperviscous simulations is given by $\nu_4 = \nu_1 k_C^{-6}$, where $k_C = \frac{N}{3}$.

For the 2D Euler equations, we consider two kinds of initial conditions. The first one is a vortex merger, made of two Gaussian vortices with positive circulation and one weaker Gaussian vortex with negative circulation [28].

The formula giving the initial vorticity is:

$$\omega(t=0, x, y) = \pi \left(e^{-\pi^2 \left(\left(x - \frac{3\pi}{4} \right)^2 + (y - \pi)^2 \right)} + e^{-\pi^2 \left(\left(x - \frac{5\pi}{4} \right)^2 + (y - \pi)^2 \right)} - \frac{1}{2} e^{-\pi^2 \left(\left(x - \frac{5\pi}{4} \right)^2 + \left(y - \pi \left(1 + \frac{\sqrt{2}}{4} \right) \right)^2 \right)} \right) \quad (8)$$

where $(x, y) \in [0, 2\pi]^2$. For the viscous simulations starting from this initial condition, we have used $\nu_1 = 13.1072N^{-2}$.

The second initial condition is a correlated Gaussian noise. To generate it, we use the Fourier representation:

$$\omega(t=0, \mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{N}^2, |\mathbf{k}| \leq k_C} \hat{\omega}[\mathbf{k}] e^{i\mathbf{k} \cdot \mathbf{x}} \quad \text{where} \quad \hat{\omega}[\mathbf{k}] = \frac{1}{4\pi^2} \int_{\mathbb{T}^2} \omega(\mathbf{x}) e^{-i\mathbf{k} \cdot \mathbf{x}} d^2\mathbf{x} \quad (9)$$

and, defining $k = |\mathbf{k}|$, we then let

$$\hat{\omega}[t=0, \mathbf{k}] = \frac{1}{24\pi^2} \times \begin{cases} k e^{i\theta_{\mathbf{k}}} & \text{if } k \leq 6 \\ 6^2 k^{-1} e^{i\theta_{\mathbf{k}}} & \text{if } 6 < k \leq 42 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

where the $\theta_{\mathbf{k}}$ are pseudo-random numbers uniformly and independently distributed in $[0, 2\pi[$. The $\theta_{\mathbf{k}}$ are drawn once using a Mersenne twister pseudo-random number generator [34], and the same values are then used for all cases studied below. The initial isotropic enstrophy spectrum

$$Z[k] = 2\pi^2 \sum_{k \leq |\mathbf{k}| < k+1} |\hat{\omega}[\mathbf{k}]|^2$$

is then approximately:

$$Z[t=0, k] \simeq \frac{1}{144\pi} \times \begin{cases} k^3 & \text{if } k \leq 6 \\ 6^4 k^{-1} & \text{if } 6 < k \leq 42 \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

The energy of this initial condition is $E(t=0) = \frac{1}{2} \langle \mathbf{u}_0 | \mathbf{u}_0 \rangle \simeq 7.847 \cdot 10^{-2}$ and its enstrophy is $Z(t=0) = \sum_{k=0}^{\infty} Z[k] = \frac{1}{2} \langle \omega_0 | \omega_0 \rangle \simeq 6.289$. For the viscous simulations starting from this initial condition, we have used $\nu_1 = 6.5536N^{-2}$.

2. RESULTS

2.1. 1D Burgers

We first briefly recall the results for the 1D Burgers equation [35], which are now confirmed by using a fully dealiased code, as described above. Compared to [35], we thus have even less numerical dissipation. An upper bound is estimated by performing simulations with $\nu = 0$ without filtering, where we observe that for $N = 4096$ the fraction of the initial energy that has been dissipated at $t = \frac{5}{\pi}$ is less than 10^{-8} .

Recall that for Burgers equation, it is possible to compute the entropy solution to (5) with a very high precision, using a Legendre transform method [6]. This procedure is particularly straightforward for the simple initial condition leading to a single shock that we have used. The obtained entropy solution will be our reference u_{ref} for this section. Note that the entropy solution dissipates energy, contrary to the Galerkin-truncated one, obtained by solving the inviscid Burgers equation using our conservative numerical scheme.

The solutions obtained with methods (i), (ii), (v) and (vi) for $N = 4096$ are shown in Fig. 1 (left). We observe that (i) and (vi) are quite similar, whereas (v) has much more pronounced oscillations. In Fig. 1 (right), the squared modulus of the Morlet wavelet transform [21] of each solution is also shown using a logarithmic colorscale, which allows us to visualize the errors in space and scale. The artifacts due to aliasing in the Coiflet

CVS filtering **(v)** are clearly visible on this figure. They are much less pronounced in the Kingslet case **(vi)**. The Gibbs phenomenon can also be noticed on the viscous **(i)** and hyperviscous **(ii)** solutions.

Two error measures allow us to quantify the difference between a numerical solution u_N and the reference entropy solution u_{ref} . The first error measure is defined as:

$$\epsilon_w = \frac{2\pi}{\|u_{\text{ref}}\|} \left(\sum_{|k| \leq 64} |\hat{u}_N[k] - \hat{u}_{\text{ref}}[k]|^2 \right)^{\frac{1}{2}} \quad (12)$$

where $\|\cdot\|$ stands for the L^2 norm. This quantity is of practical interest, since it frequently occurs that only the low wavenumber behavior of the solution needs to be accurately described. From the mathematical point of view, it can be seen as an indicator of weak convergence. The second error measure is the total L^2 error:

$$\epsilon = \frac{\|u_N - u_{\text{ref}}\|}{\|u_{\text{ref}}\|} \quad (13)$$

which measures strong convergence to the entropy solution.

Our first result is that in these simulations, weak convergence of the Galerkin-truncated inviscid simulations to the entropy solution does not occur (Fig. 2, left), in agreement with a remark made by Tadmor in [44]. This confirms, if need be, that regularization is indispensable to solve Burgers equation with a spectral method. Post-processing the result of an unregularized simulation does not yield a physically relevant solution. On the contrary, the solutions given by all other methods appear to converge weakly to the entropy solution with a rate $O(N^{-1})$. The error associated to the viscous and Kingslets solutions is about one order of magnitude smaller than the error associated to the Coiflet solution.

The solutions obtained with every considered regularization methods also converge strongly towards the entropy solution with a rate $O(N^{-1/2})$ (Fig. 2, middle). This confirms the results of [35] and extends them to several new cases, namely real-valued wavelets and linear filtering. As a sidenote, we confirm that the hyperdissipative 1D Burgers model does not improve in any way the results compared to the viscous model [19].

In Fig. 2 (right), we have plotted the compression rate, defined as follows:

$$\rho = \frac{N}{N_C} \quad (14)$$

where $N_C = N - N_I$ is the number of wavelet coefficients above the threshold. The subscripts C and I stand respectively for coherent and incoherent. The higher the compression rate, the better the compression.

The following conclusions may be drawn from the results in this section. To regularize the 1D inviscid Burgers equation, it is enough to filter out the wavelet coefficients at the finest scale. The results are better for complex-valued wavelets because of their reduced aliasing effects. The main advantage of CVS wavelet filtering compared to other regularization methods is that they allow us to compress the information needed to represent the solution. Here, we observe that the compression rate ρ is about 3 for $N = 1024$ and improves slowly for higher resolutions, a fact that we will comment more on in the conclusion.

2.2. 2D Euler

As in the previous subsection, we study the results obtained with methods **(i)**-**(vi)**, but this time to solve the 2D Euler equations (3). Five increasingly refined resolutions were used in each case: $N = 2^7, \dots, 2^{11}$. Since we do not have an efficient way of computing the solution to (3) with high precision, we use the viscous simulation **(i)** with $N^2 = 8192^2$ as a reference. Inviscid simulations at $N^2 = 2048^2$ show that the fraction of the initial enstrophy dissipated during the whole simulation is less than $5 \cdot 10^{-2}$, and less than 10^{-5} for the energy. Note that the numerical dissipation is due to the error coming from the time discretization of the equations, and should therefore be smaller in the viscous and filtered simulations, since the corresponding solutions are smoother in time.

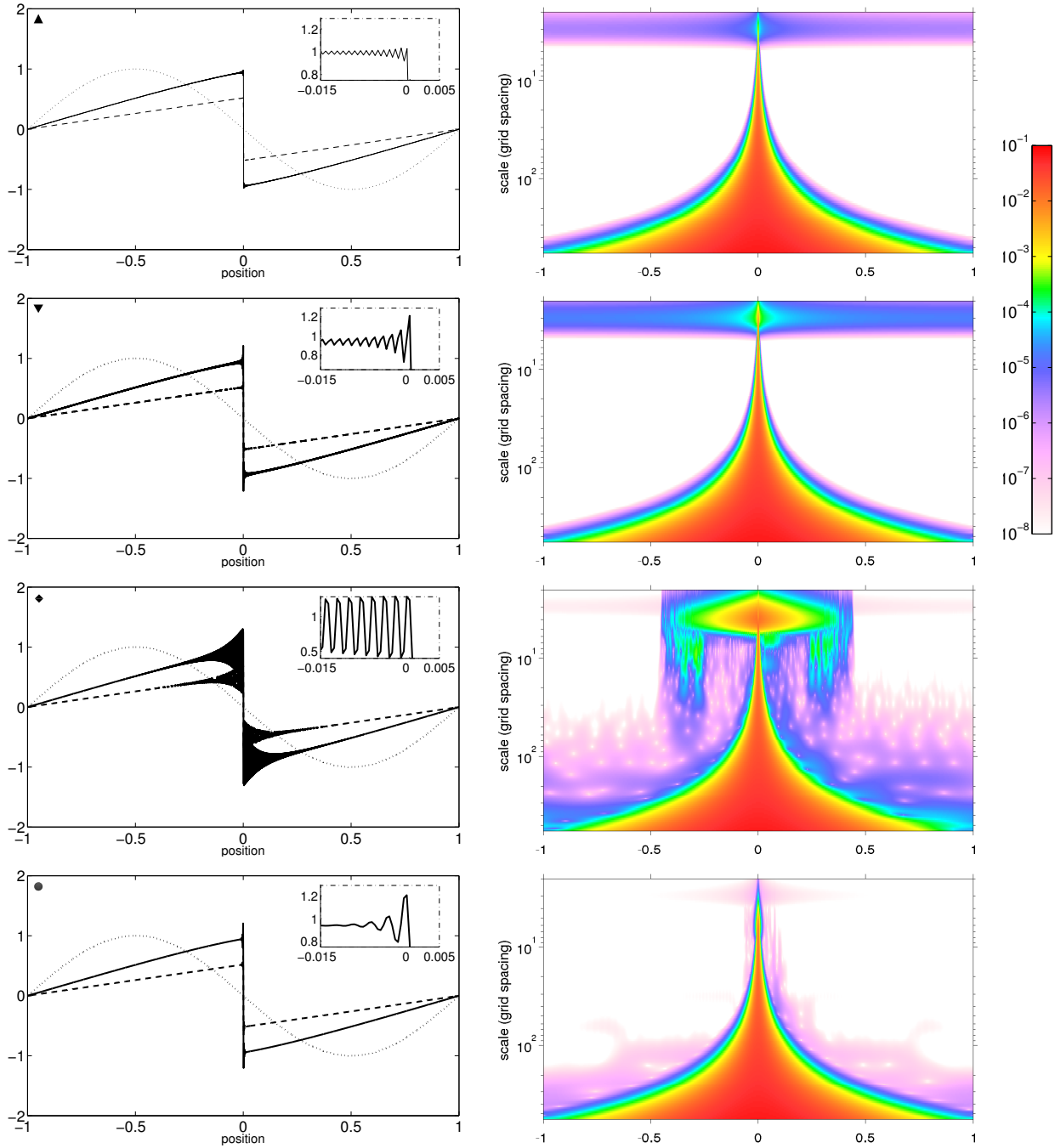


FIGURE 1. Left column: numerical solution to the 1D Burgers equation for methods (i), (ii), (v) and (vi) (top to bottom) with $N = 4096$. Dotted lines: $t = 0$, solid lines: $t = \frac{2}{\pi}$, dashed lines: $t = \frac{5}{\pi}$. The insets show a zoom around $x = 0$. Right column: squared modulus of the Morlet coefficients of each solution at $t = \frac{5}{\pi}$. The vertical scale is logarithmic and indicates the wavelet scale factor, normalized by the distance between two grid points. Note that the color scale is also logarithmic.

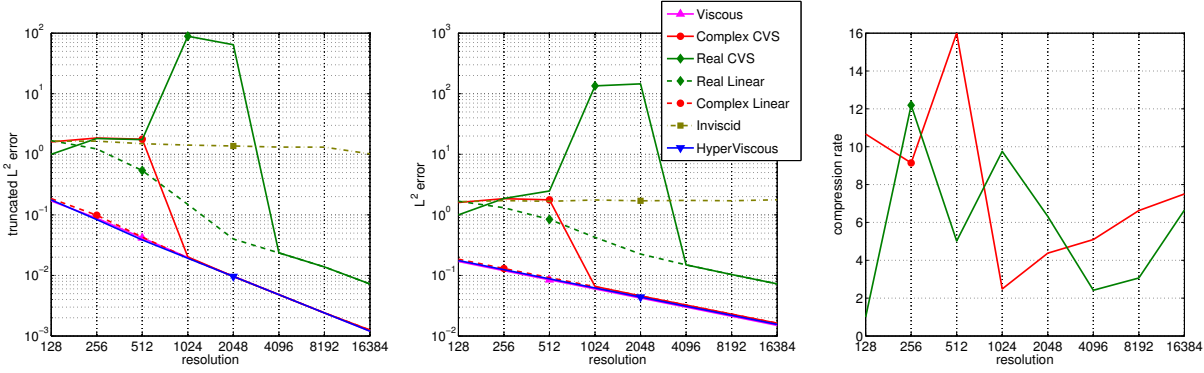


FIGURE 2. Comparison between the six regularization methods to solve the 1D Burgers equation. Left: normalized L^2 -error for the Fourier modes with $|k| \leq 64$. Middle: total normalized L^2 -error. Right: Compression rate $\rho = \frac{N}{N_C}$.

2.2.1. Deterministic initial condition

In this subsection, we consider the vortex merger initial condition (8). The vorticity field at $t = 50$ for methods (i), (ii), (v) and (vi) are shown in Fig. 3. The reference vorticity field (top, left) and the result of the inviscid simulation (bottom, right) are also shown for comparison purposes. The gross dipolar structure of the vorticity field is well preserved by all methods. The main differences have to do with sharp gradients that are generated very rapidly during the merging of the two positively signed vortices. Each positive vortex corresponds to a nonuniform velocity field that causes the other positive vortex to rotate around it and deform, producing a spiral pattern that is still visible in Fig. 3. These sharp gradients are clearly not very well handled by method (v), since artifacts have been introduced even in quiet regions of the domain.

The different regularization mechanisms induce a different behavior of the solution in a global sense, as can be seen on the time evolutions of energy and enstrophy, and on the enstrophy spectra (Fig. 4).

We now examine the difference between all the solutions and the reference solution in a quantitative manner. In Fig. 5 (left), we plot the truncated L^2 -error, which is defined by (12) with an additional 2π factor due to normalization. These curves illustrate that even the low wavenumber dynamics are affected by the regularization mechanism. Nevertheless, the error seems to converge to zero for all considered regularization methods. The rate is $O(N^{-2})$ for the viscous (i) simulations, consistent with known analytical results concerning the inviscid limit of Navier-Stokes equations [26]. The arguments in [26] apply equally well to the hyperviscous Navier-Stokes equations, and we accordingly observe the same convergence rate for (ii). The rate is less easy to determine for the other methods, but it seems to lie between $O(N^{-2})$ and $O(N^{-1})$.

Now the total L^2 -error, defined similarly to (13) above, is represented in Fig. 5 (right). The results are strikingly close to the previous ones, indicating that most of the L^2 -error in fact comes from low wavenumbers. The results concerning the inviscid Galerkin-truncated Euler solution (square marker) are particularly interesting since they contrast with those observed for the 1D Burgers equation above. It was previously observed in [9] “that the spectrally truncated [3d] Euler equations have long-lasting transients behaving just like those of the dissipative [3d] Navier-Stokes equations”. This result is also supported by calculations using the eddy-damped quasi-normal Markovian closure theory [5]. Our results are unfortunately limited to two space dimensions, but they are stronger, since they indicate that the solution to the Galerkin-truncated 2D Euler equations converges to the solution of Euler equations in the L^2 norm.

By looking at the numbers presented in this section, one could draw the hasty conclusion that hyperviscous or even inviscid simulations do the job very well, so why bother with wavelet filtering ? But as we have already

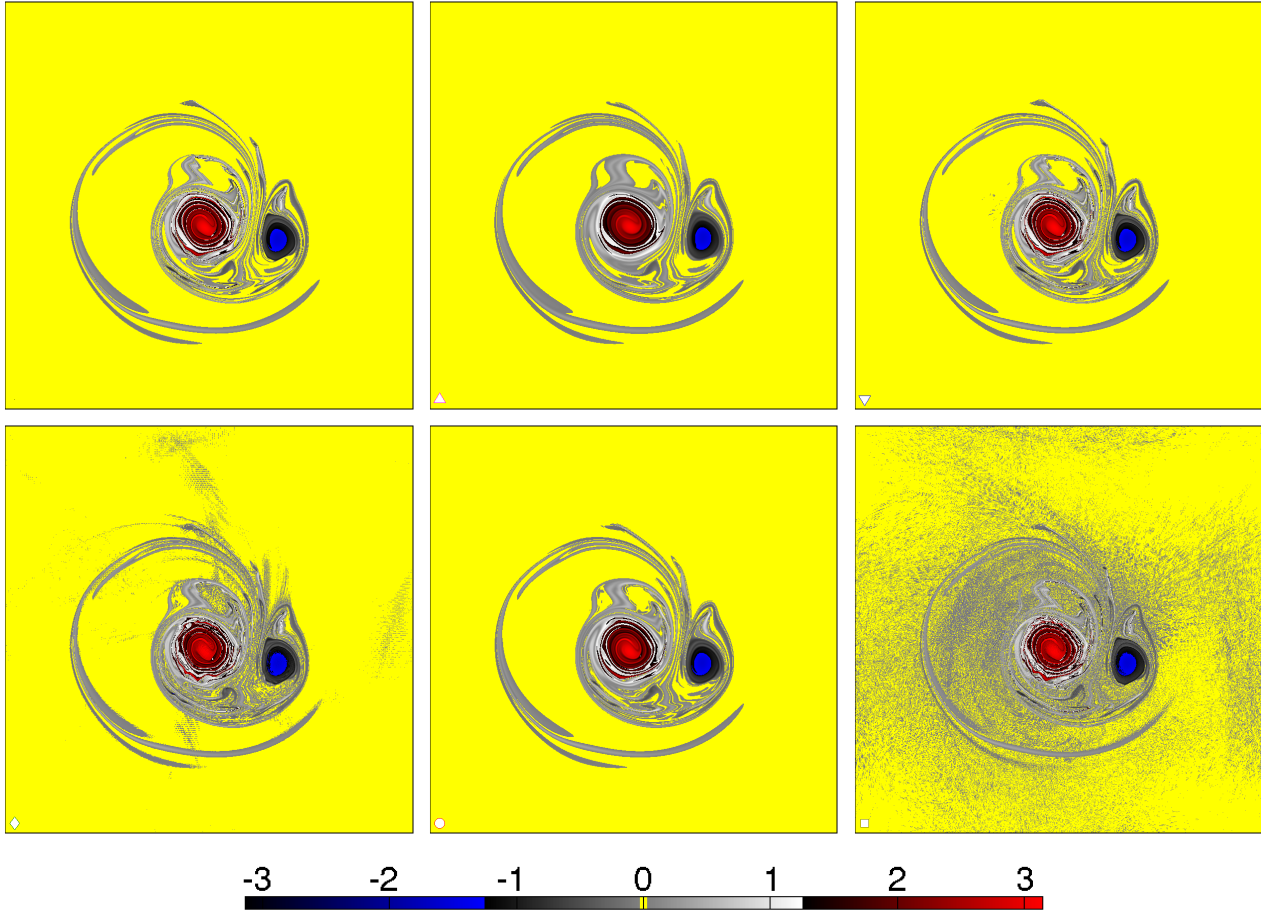


FIGURE 3. Vorticity fields at $t = 50$ for the deterministic initial condition. Top left: reference Navier-Stokes solution, $N = 8192$. Top middle: Navier-Stokes solution (i), $N = 2048$. Top right: Hyperdissipative Navier-Stokes solution (ii), $N = 2048$. Bottom left: Coiflet CVS filtered Euler solution (v), $N = 2048$. Bottom middle: Kingslets CVS filtered Euler solution (vi), $N = 2048$. Bottom right: Galerkin-truncated Euler solution, $N = 2048$.

seen for Burgers equation, the advantage of wavelets has to do with compression. We now turn to random initial conditions, in order to demonstrate this for Euler equations.

2.2.2. Random initial condition

By computing the flow evolution starting from the randomly generated initial condition (10), we obtain the vorticity field at $t = 50$ for methods (i), (ii), (v) and (vi), shown in Fig. 6. As above, the reference vorticity field (top, left) and the result of the inviscid simulation (bottom, right) are also given. Only the subdomain $[\pi, 2\pi] \times [\pi, 2\pi]$ is represented on the figure, in order to make the details more visible. The first observation that can be put forward is that the positions of all vortices match pretty well between all represented fields. Of course, this situation is bound not to last for long, since fully developed 2D turbulence is sensitive to initial conditions: the different simulations should separate sooner or later and end up having nothing left in common. But this occurs on a longer time scale than the one we have considered here. Note that the initial eddy turnover time $\tau := \frac{1}{\sqrt{2Z(t=0)}}$ is approximately 0.282 here.

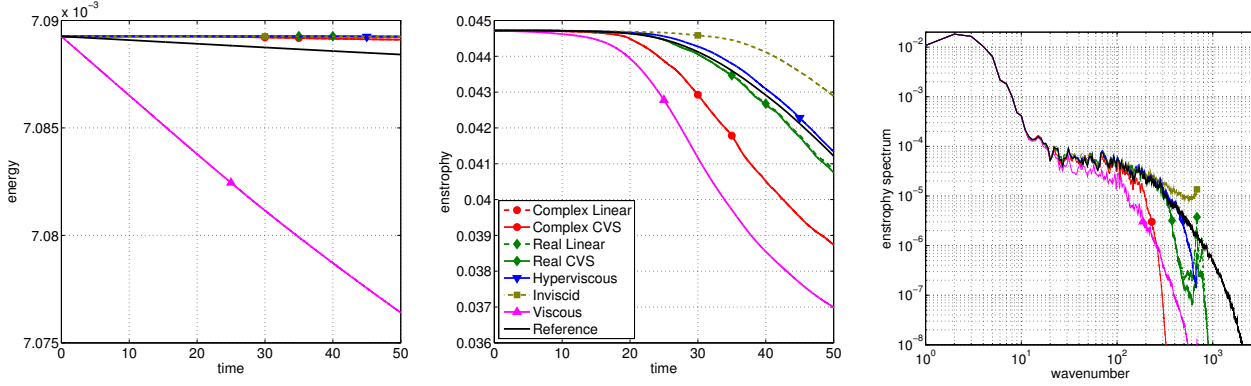


FIGURE 4. Deterministic initial condition: results for the different methods with $N = 2048$, and reference viscous solution with $N = 8192$. Left: time evolution of energy. Middle: time evolution of enstrophy. Right: enstrophy spectra at $t = 50$.

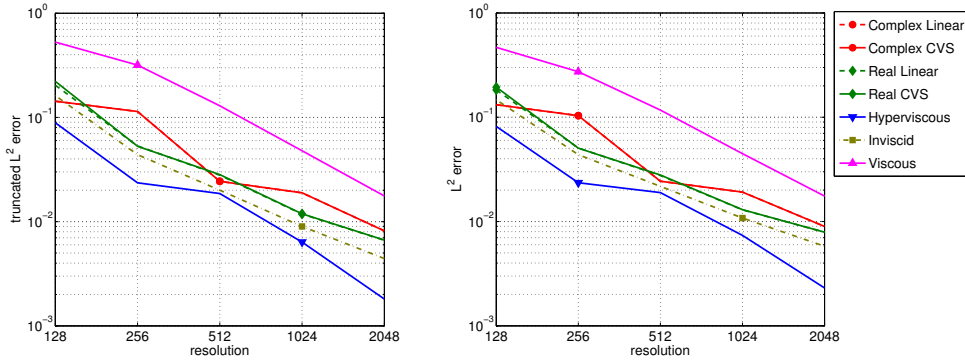


FIGURE 5. Left: truncated L^2 error $\epsilon_w(12)$ at $t = 50.05$ with respect to the reference solution. Right: total L^2 error $\epsilon(13)$ at $t = 50.05$ with respect to the reference solution.

We would like to outline two kinds of differences between the fields yielded by the different methods. First, their high wavenumber behavior is quite different. This can be checked on the enstrophy spectra (Fig. 8, right). In this respect, the similarity between the Kingslet-filtered (**vi**) (bottom, middle) and viscous (top, middle) vorticity fields is striking. Due to its better Fourier localization of the dissipation term, the hyperviscous simulation reproduces even better the high wavenumber features of the reference simulation, as is highlighted by the cuts in Fig. 7.

The second difference can be seen in Fig. 8 (left) and concerns only regularization by CVS filtering using Coiflets (**v**), for which energy starts increasing again after $t \simeq 60$, contrary to all other methods. This effect suggests that the flow has been affected by the regularization method in an unphysical way, and was our main motivation for introducing Kingslets in this study as a better alternative.

The time evolution of enstrophy is shown in Fig. 8 (middle). Even the reference simulation is still quite dissipative concerning enstrophy. This observation is consistent with a very slow decay of the enstrophy dissipation rate as a function of Reynolds number [12, 32, 45], which implies a slow convergence to the Euler solution in the H^1 norm. Note that for the viscous solution, convergence is known to occur at a rate $O(\nu) = O(N^{-2})$ in any

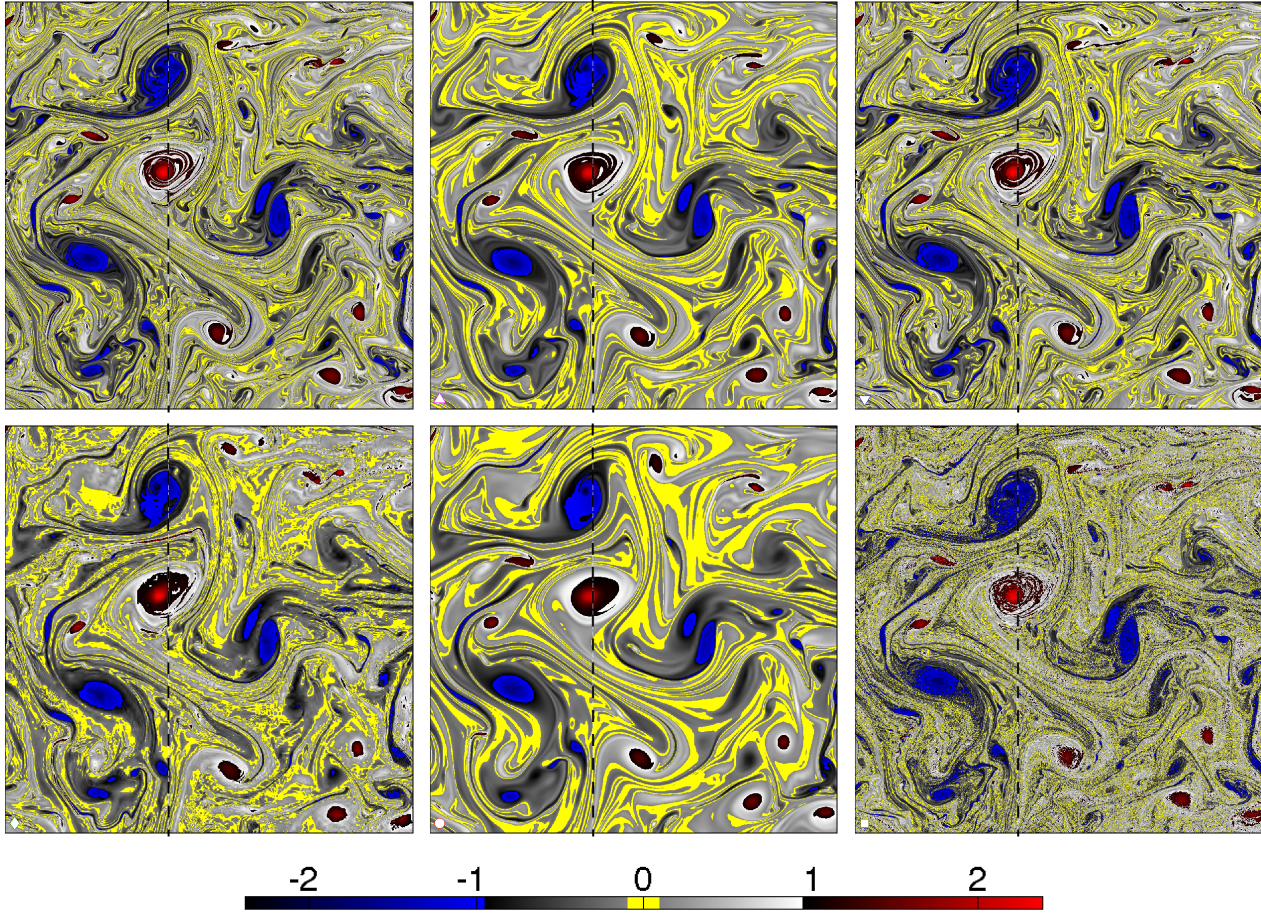


FIGURE 6. Vorticity fields at $t = 50$ for the random initial condition, restricted to the square $[\pi, 2\pi] \times [\pi, 2\pi]$. Top, left: reference Navier-Stokes solution with $N = 8192$. Top, middle: Navier-Stokes solution (i), $N = 2048$. Top, right: Hyperdissipative Navier-Stokes solution (ii), $N = 2048$. Bottom, left: Coiflet CVS filtered Euler solution (v), $N = 2048$. Bottom, middle: Kingslets CVS filtered Euler solution (vi), $N = 2048$. Bottom, right: Galerkin-truncated Euler solution, $N = 2048$. The black dashed lines indicates the location of the cuts shown in Fig. 7.

norm H^m [26], since the initial condition considered here consists in a finite number of Fourier modes and is therefore analytic. We believe that this rate is impossible to observe numerically because the constant in front of ν grows extremely fast in time. Although it would be interesting to consider also the H^1 -error, we conclude that this quantity is out of our reach here. Indeed, because of the slow convergence, the viscous solution at $N = 8192$ cannot be used as a reference to compute the H^1 norm error between (i)-(vi) and the Euler solution, nor can any solution computed with current day methods and computational power.

In Fig. 8 (right), the enstrophy spectra of the different solutions are compared with the one of the reference. They all agree well at low wavenumbers, as can be seen in the inset.

Finally, we study the convergence in the inviscid limit, for fixed time and fixed initial condition (Fig. 9). As in the case of the deterministic initial condition, the truncated L^2 -error ϵ_w and the total L^2 -error ϵ are represented. The error decay is less pronounced in the considered range of Reynolds number than it was in

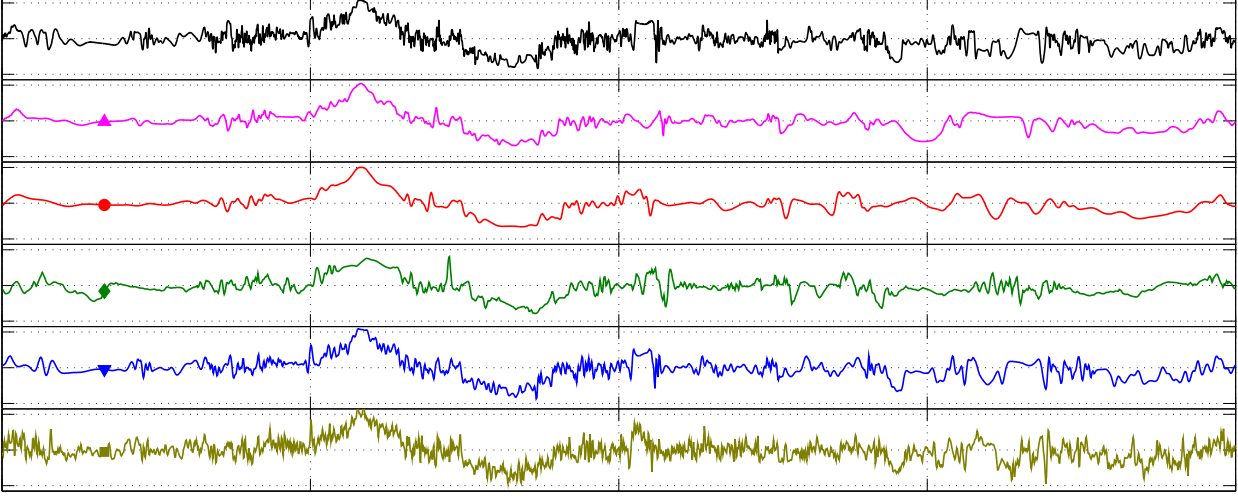


FIGURE 7. Cuts through the vorticity fields at $x = \frac{7\pi}{5}$ and $t = 50$, for the different methods with $N = 2048$, and for the reference viscous solution with $N = 8192$. The legend can be found in Fig. 8.

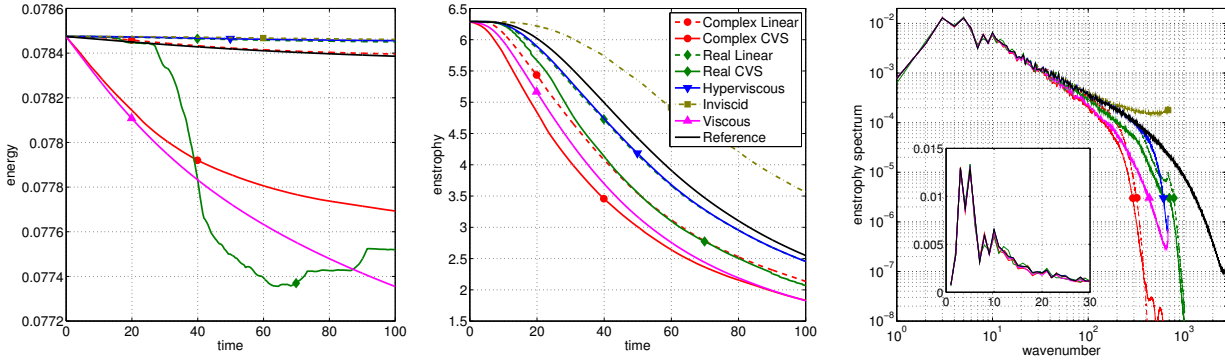


FIGURE 8. Random initial condition, advanced with different methods for $N = 2048$, and reference viscous solution for $N = 8192$. Left: time evolution of energy. Middle: time evolution of enstrophy. Right: enstrophy spectra at $t = 50$. The inset shows the low wavenumber part in lin-lin coordinates. The legend applies to all three graphs.

the deterministic case above. It seems that the Reynolds number at which the analytically predicted $O(N^{-2})$ asymptotic behavior manifests itself has not been reached yet.

We now comment on the compression rate ρ (14), plotted in Fig. 9, right. One can see that the CVS thresholding allows us to compress by a factor of about 3 using Kingslets, and about 10 using Coiflets. The computations presented here do not benefit from this compression, since the solution is reconstructed in Fourier space at each timestep. Nevertheless, the measured compression rate remains a good indicator of what can be achieved by adaptive wavelet-based solvers [15, 41] using the thresholding rule that we have defined earlier.

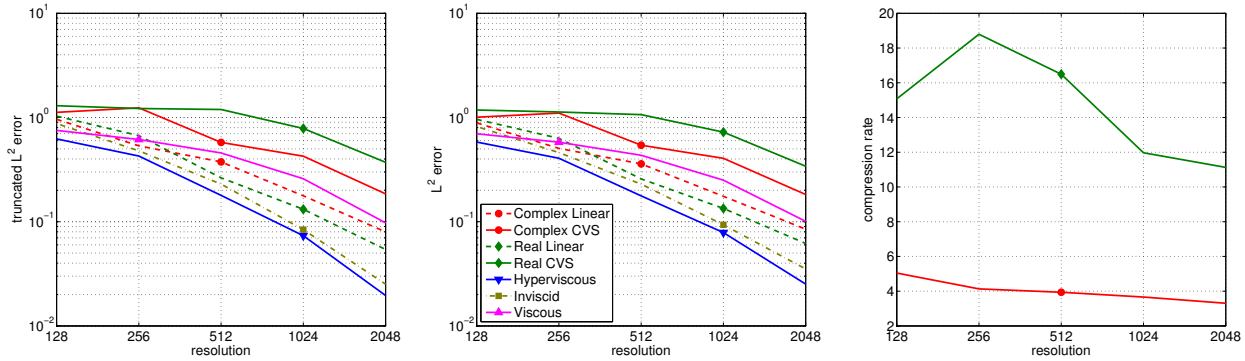


FIGURE 9. Random initial condition. Left: truncated L^2 -error ϵ_w (12) with respect to the reference solution. Middle: total L^2 -error ϵ (13) with respect to the reference solution. Right: compression rate (14) achieved by nonlinear wavelet thresholding. The legend applies to all three graphs.

3. CONCLUSION AND PERSPECTIVES

We have used a classical Fourier-Galerkin method to solve the 2D Euler equations, and we have proposed a new wavelet-based approach to regularize the solution, in order to obtain a good numerical approximation of the Euler solution. We have compared our method with viscous and hyperviscous regularization, and also with the solution computed without regularization. The ability of all regularization methods to approach the inviscid dynamics was measured quantitatively by direct comparison with a high resolution viscous simulation. High resolution numerical simulations were made possible thanks to parallel implementations of the spectral code and of the wavelet transform.

The main findings concerning the 2D Euler equations can be summarized as follows:

- linear wavelet filtering approaches the inviscid dynamics as well as viscous or hyperviscous dissipative terms,
- nonlinear wavelet filtering with real-valued wavelets (Coiflets) introduces undesirable artifacts in the solution,
- nonlinear wavelet filtering with complex-valued wavelets (Kingslets) preserves the dynamics and offers at the same time a non negligible compression rate of about 3 for fully developed turbulence,
- there are strong indications that the solution to the Galerkin-truncated 2D Euler equations converges, at least in the weak sense, to the solution to the 2D Euler equations.

We can hence claim that the main goals of our wavelet regularization method have been attained. Indeed, we have shown that CVS filtering with Kingslets allows us to regularize the 2D Euler equations, and at the same time compress the solution. This is a strict extension of the results obtained for the 1D Burgers equation. One possible explanation for the disappointing results observed with Coiflets is the lack of a safety zone in wavelet space [40]. Since the orthogonal wavelet transform is not translation invariant, it cannot properly capture structures that are advected in a continuous fashion by the flow. The rather low compression rates observed when filtering solutions of the 1D Burgers equation should be seen as a limitation of the employed thresholding method, and not of the wavelets themselves. Indeed, for a function as simple as single shock, wavelets are in principle able to achieve much larger compression rates. Improving the threshold selection algorithm is a priority goal of our ongoing research work.

We would now like to draw attention to the striking differences between the results that we found for the 1D Burgers equation on the one hand and for the 2D incompressible Euler equations on the other hand. Indeed, the

Galerkin-truncated dynamics of these two equations behave very differently in comparison to their non truncated counterparts: for the 2D Euler equations we have shown weak convergence of the solution to the Galerkin-truncated equations towards the inviscid solution, while for the 1D Burgers equation no such convergence was observed, as Tadmor has shown using an elegant mathematical argument [44]. Therefore, one should be extremely cautious when using Burgers equation as a toy model for understanding incompressible turbulence. One may argue that the 2D Euler dynamics is pathological since it does not have finite time singularity (FTS), and that Burgers is in this respect closer to 3D Euler. But even if the 3D Euler equations have FTS, these singularities cannot be of the same nature than the shocks occurring in the solutions of Burgers equation. The incompressibility condition plays a key role by introducing nonlocal effects that are completely absent in the Burgers model. To illustrate these differences further, consider the results obtained with hyperviscous regularization. In [19], it was argued that hyperviscosity introduces a quasi-thermalized range in the spectrum. As a consequence, all that is gained by the better wavenumber localization of the dissipation term is lost because of an enhanced bottleneck. These results were supported by 1D Burgers direct numerical simulations, and by the eddy-damped quasi-normal Markovian (EDQNM) closure for 3D turbulence. Our results are in agreement concerning 1D Burgers, but show that hyperviscous regularization offers very good performance for 2D Euler, as suggested by previous results [28]. It cannot be decided at present if the discrepancy is due to a shortcoming of the EDQNM closure, or if the hyperviscous approach breaks down when going from 2D to 3D. Checking this via direct numerical simulation in the 3D case is an interesting topic for future research.

In future work, we would also like to undertake the same kind of study with an adaptive wavelet-Galerkin solver instead of a Fourier-Galerkin solver. This will allow us to benefit at the same time from the compression and from the regularization offered by the CVS wavelet filter, and to measure the speed-up. Encouraging results concerning compression are already available in the literature on adaptive wavelet methods, see e.g. [41] and references therein.

Acknowledgements: We are very grateful to Nick Kingsbury for his help in developing the parallel wavelet transform, and his renewed hospitality in Trinity College, Cambridge, U.K. and in the Engineering Department, Cambridge University, U.K. We would also like to thank Claude Bardos, Uriel Frisch, Isabelle Gallagher, Thierry Paul and Laure Saint-Raymond for fruitful mathematical discussions. MF is very grateful to both Trinity College, Cambridge, and the Wissenschaftskolleg zu Berlin for hospitality.

4. APPENDIX: PARALLEL WAVELET TRANSFORM ALGORITHM

To efficiently filter the solution at each timestep of the simulation, the wavelet transform should be performed in-place and integrated within the simulation loop. Since solving the 2D Navier-Stokes equations is much more computer intensive than solving the 1D Burgers equation, we have faced the necessity of implementing the wavelet transform and CVS filtering algorithms in a parallel computation environment, like the spectral solver itself.

The parallelization of the fast wavelet transform algorithm was first demonstrated in architecture dependent frameworks [24], but since then much of the effort has been spent in developing efficient algorithms that run within generic distributed memory parallel environments such as MPI or PVM. Early approaches used to mimic the parallel FFT algorithm, which involves a global transposition step that requires exchange of data between all processes, and thus strongly degrades efficiency. The parallelization strategy that we employ here was first proposed and studied in [36] for both the 1D wavelet transform and the 2D tensor-product wavelet transform. The key ingredient to obtain good scalability is the domain decomposition approach, since it preserves the quasi-locality of the wavelet transform. It was extended to 2D multiresolution analyses in [8], who reported parallelization efficiency of up to 0.9 using a maximum of 32 processes and images of size up to 2048×2048 .

Our objective in this section is to measure the efficiency of this parallelization method for more challenging problem sizes, and also in three dimensions. For this, we introduce decomposition of the domain along all available dimensions instead of just one, which allows us to obtain good scalability in three and higher dimensions.

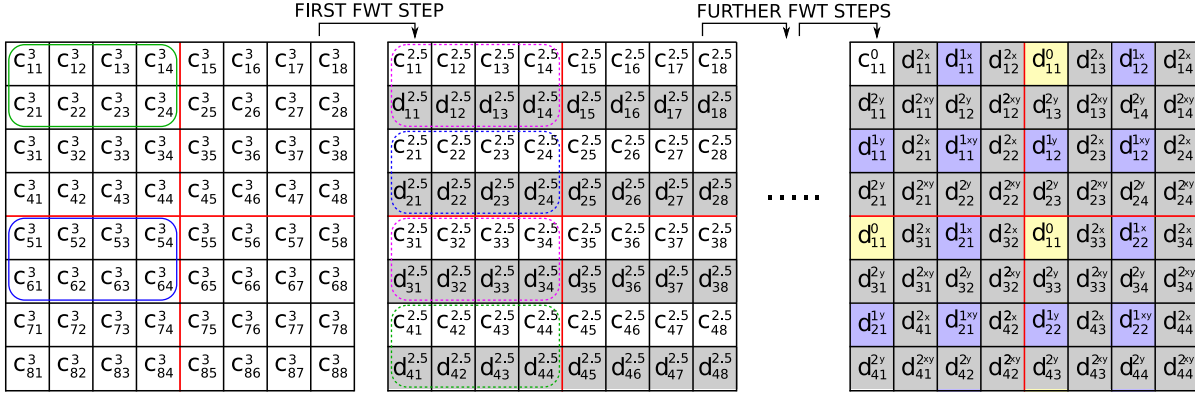


FIGURE 10. Schematic view of the ordering of the wavelet coefficients before and after transformation. Red lines: possible domain decomposition for distributed memory parallelization involving 4 processes. Round edged boxes: communication between processes to compute the first step (see text).

Note that [8] and other references also discussed key issues related to local memory bandwidth optimization, which will not be elaborated upon since our main concern is parallelization.

We first establish some notations concerning the wavelet transform. The transform is assumed to be real and orthogonal. In the complex case, the notations remain the same, with exponents R and I added to distinguish the real and imaginary trees. Let $h[n]$ and $g[n]$ be respectively the scaling and wavelet quadrature mirror filters. We assume that they are nonzero only for $n = 0, \dots, S-1$. The 1D fast wavelet transform algorithm [33] relates the coefficients at scale j and $j-1$ via:

$$c_p^{j-1} = \sum_{n=0}^{S-1} h_n c_{2p+n}^j \quad (15)$$

$$d_p^{j-1} = \sum_{n=0}^{S-1} g_n c_{2p+n}^j \quad (16)$$

and the N -dimensional wavelet transform is computed by applying (15-16) with the position p varying successively along each direction.

Let us momentarily take only one process and describe how the data is arranged in memory. The index which varies most rapidly is assumed to correspond to the first array dimension (Fortran ordering). For illustrative purposes we restrict our attention to a 2D array of size 8×8 . The memory locations of the scaling function coefficients at the finest scale are shown in Fig. 10 (left). The red lines should be ignored for now. The FWT algorithm starts by computing a single level of the filter bank on the columns of the matrix, using (15) and (16). The results of the computation are stored in-place, alternating between scaling coefficient and wavelet coefficient. Due to the periodic wrapping of the column, the leading $S-2$ coefficients at scale J are needed to compute the trailing $S-2$ coefficients at scale $J-1$. These coefficients should therefore not be overwritten right away, but held in memory until they are not needed anymore. This can be accomplished thanks to a small buffer of size $S-2$. No additional memory is required. The results of this very first step are shown in Fig. 10 (middle). The exponents 2.5 indicate that we stand halfway through the first level of the transform. Once all the columns have been filtered comes the turn of the lines. They are processed in exactly the same way, except that the stride in memory is now much larger, which makes this step usually more expensive due to cache issues. After all array dimensions have been filtered, the input data has been replaced by scaling and wavelet coefficients at

scale $J - 1$. The scaling coefficients sit in the memory locations with even-numbered row and column indices. The remaining memory locations are filled with the wavelet coefficients at scale $J - 1$, which do not need to be touched by any further steps of the algorithm. The next level in the filter bank can hence be computed exactly in the same way as the first, as long as only the even-numbered rows and columns of the original memory space are considered. Continuing this procedure until only one row and one column remain yields the fully wavelet transformed data, arranged as shown in Fig. 10 (right).

Now consider that the initial data is split between 4 processes, as indicated by the red lines in Fig. 10 (left). Processes are numbered in row-major order. We assume that the sizes of the subdomains belonging to each process are equal and are powers of 2. The wavelet transform is done exactly in the same way as above, except that some interprocess communication is now necessary every time a filter needs data belonging to another process. This communication can be partially overlapped with computation in the following way. Imagine that at a certain stage in the transformation one wants to apply the filter (h_n) to the lines of a distributed matrix. This corresponds to the first step outlined in Fig. 10. From formula (15), it appears that only the $S - 2$ leading coefficients of each column need to be transmitted to another process. Before starting the computation, calls to the nonblocking point to point routines `MPI_Isend` and `MPI_Irecv` are made to initiate transmission. For example, assume momentarily that $S = 4$. To compute the coefficients in the dotted blue box (Fig. 10), all the coefficients in the solid blue box need to be sent from process 2 to process 1. Because of periodicity, all the coefficients in the solid green box need to be sent from process 1 to process 2. The same kind of communication occurs between processes 3 and 4 during this step. The coefficients in the magenta boxes can be computed without waiting for the communications to be completed. For large array sizes, they will correspond to the majority of coefficients, ensuring a good overlap of computation and communication.

Finally, we should explain how the multiple trees in the DTCWT are handled. For a 2D DTCWT transform, there are in fact 4 real wavelet transform trees [30]. We want to alternate the coefficients of the 4 trees in the same memory space. To do that, the input data is first copied into a four times larger memory space. Each input value is copied twice consecutively, and two locations are left uninitialized between values. The wavelet transforms at the finest scale of the columns of the two intricately submatrices that have been initialized are performed in-place using the appropriate filters [30]. The results are then duplicated to fill in the blank locations. The appropriate filters are then applied to the lines of the four intricately submatrices. The remaining scales are computed normally. Parallelisation follows straightforwardly using the same method as above.

The consumed CPU time for a forward wavelet transform using the algorithm just described, with Daubechies filters of length 4, was measured on an IBM Regatta Power6 machine. The results are shown in Fig. 11. One observes near perfect speedup up to 32 processes. The sudden degradation of efficiency when going from 32 to 64 processes is mostly due to hardware limitations, since each node on the machine contains 32 Power6 cores. Simulations with a higher number of cores require communication through a different network, which has a longer latency. We have also performed simulations on a BlueGene/P machine, which show good scaling results for 3D arrays using up to 1024 processes. These tests were done with arrays of sizes 1024^3 . They will be reported in detail elsewhere.

A C++ implementation of this algorithm is available under the GNU General Public License at the following URL: <https://sourceforge.net/project/kicksey-winsey>.

REFERENCES

- [1] Alexandre Azzalini, Marie Farge, and Kai Schneider. Nonlinear wavelet thresholding: a recursive method to determine the optimal denoising threshold. *Applied Computational Harmonic Analysis*, 18(2):177–185, October 2004.
- [2] Claude Bardos, Jasmine S. Linshiz, and Edriss S. Titi. Global regularity for a Birkhoff-Rott- α approximation of the dynamics of vortex sheets of the 2d Euler equations. *Physica D: Nonlinear Phenomena*, 237(14–17):1905–1911, 2008.
- [3] Claude Basdevant, Bernard Legras, Robert Sadourny, and M. B  land. A study of barotropic model flows: intermittency, waves and predictability. *Journal of the Atmospheric Sciences*, 38:2305–2326, 1981.
- [4] G.K. Batchelor. *The Theory of Homogeneous Turbulence*. Cambridge University Press, 1953.
- [5] W. J. T. Bos and J.-P. Bertoglio. Dynamics of spectrally truncated inviscid turbulence. *Physics of Fluids*, 18(7):071701, 2006.
- [6] J. M. Burgers. *The nonlinear diffusion equation: asymptotic solutions and statistical problems*. D. Reidel Pub. Co., 1974.

- [7] Claudio Canuto, Alfio Quarteroni, M. Yousuff Hussaini, and Thomas A. Zang. *Spectral methods in fluid dynamics*. Springer-Verlag, 1988.
- [8] D. Chaver, M. Prieto, L. Pinuel, and F. Tirado. Parallel wavelet transform for large scale image processing. In M. Prieto, editor, *Parallel and Distributed Processing Symposium., International, IPDPS 2002*, pages 4–9, 2002.
- [9] Cyril Cichowlas, Pauline Bonaïti, Fabrice Debbasch, and Marc Brachet. Effective dissipation and turbulence in spectrally truncated Euler flows. *Physical Review Letters*, 95(26):264502, Dec 2005.
- [10] R. R. Coifman and D.L. Donoho. Translation-invariant denoising. pages 125–150. Springer-Verlag, 1995.
- [11] Ingrid Daubechies. Orthonormal bases of compactly supported wavelets ii. variations on a theme. *SIAM Journal of Mathematical Analysis*, 24(2):499–519, 3 1993.
- [12] Pablo Dmitruk and David C. Montgomery. Numerical study of the decay of enstrophy in a two-dimensional Navier-Stokes fluid in the limit of very small viscosities. *Physics of Fluids*, 17:035114, 2005.
- [13] Gregory L. Eyink and Katepalli R. Sreenivasan. Onsager and the theory of hydrodynamic turbulence. *Reviews of Modern Physics*, 78(1):87, 2006.
- [14] Marie Farge, Eric Goirand, Yves Meyer, Frédéric Pascal, and Mladen Victor Wickerhauser. Improved predictability of two-dimensional turbulent flows using wavelet packet compression. *Fluid Dynamics Research*, 10(4-6):229 – 250, 1992.
- [15] Marie Farge and Kai Schneider. Coherent vortex simulation (CVS), a semi-deterministic turbulence model using wavelets. *Flow, Turbulence and Combustion*, 66(4):393–426, 2001.
- [16] Marie Farge, Kai Schneider, and Nicholas Kevlahan. Non-gaussianity and coherent vortex simulation for two-dimensional turbulence using an adaptative orthogonal wavelet basis. *Physics of Fluids*, 11(8):2187–2201, 1999.
- [17] Matteo Frigo and Steven G. Johnson. The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005. Special issue on “Program Generation, Optimization, and Platform Adaptation”.
- [18] Uriel Frisch. *Turbulence: the legacy of A.N. Kolmogorov*. Cambridge University Press, 1995.
- [19] Uriel Frisch, Susan Kurien, Rahul Pandit, Walter Pauls, Samriddhi Sankar Ray, Achim Wirth, and Jian-Zhou Zhu. Hyperviscosity, Galerkin truncation and bottlenecks in turbulence. *Physical Review Letters*, 101:144501, 2008.
- [20] D. Gottlieb and J. S. Hesthaven. Spectral methods for hyperbolic problems. *Journal of Computational and Applied Mathematics*, 128(1–2):83–131, 2001.
- [21] A. Grossmann and J. Morlet. Decomposition of Hardy functions into square integrable wavelets of constant shape. *SIAM Journal of Mathematical Analysis*, 15(4):723–736, 1984.
- [22] Ami Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 49(3):357 – 393, 1983.

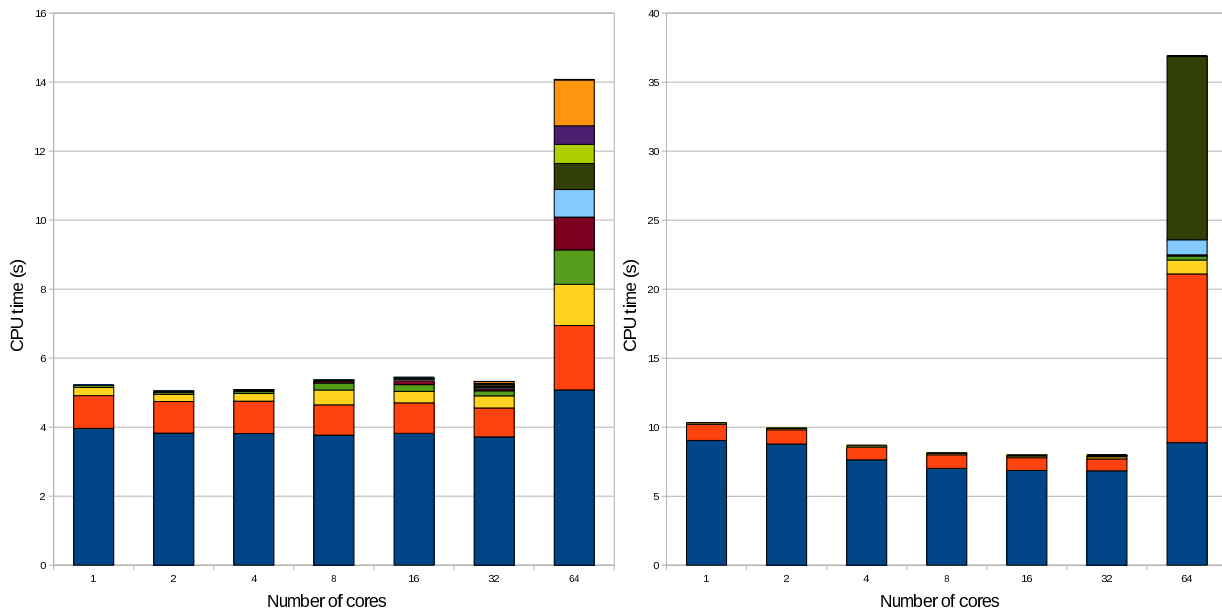


FIGURE 11. Consumed CPU time for increasing number of MPI processes on the IBM Regatta Power6. Left: 4096×4096 2D array. Right: $256 \times 256 \times 256$ 3D array. The different colors indicate successive scales in the wavelet transform, starting from the finest scale at the bottom.

- [23] Darryl D. Holm, Jerrold E. Marsden, and Tudor S. Ratiu. Euler-Poincaré models of ideal fluids with nonlinear dispersion. *Physical Review Letters*, 80(19):4173–4176, May 1998.
- [24] Mats Holmström. Parallelizing the fast wavelet transform. *Parallel Computing*, 21:1837–1848, 1995.
- [25] Yuiukio Kaneda, Takashi Ishihara, Mitsuo Yokokawa, Ken’ichi Itakura, and Atsuya Uno. Energy dissipation rate and energy spectrum in high resolution direct numerical simulations of turbulence in a periodic box. *Physics of Fluids*, 15(2):L21–L25, 2003.
- [26] T. Kato. Nonstationary flows of viscous and ideal fluids in \mathbb{R}^3 . *Journal of Functional Analysis*, 9:296–305, 1972.
- [27] Tosio Kato. On classical solutions of the two-dimensional non-stationary Euler equation. *Archive for Rational Mechanics and Analysis*, 25(3):188–200, January 1967.
- [28] Nicholas Kevlahan and Marie Farge. Vorticity filaments in two-dimensional turbulence: creation, stability and effect. *Journal of Fluid Mechanics*, 346:49–76, 1997.
- [29] Boualem Khouider and Edriss S. Titi. An inviscid regularization for the surface quasi-geostrophic equation. *Communications on Pure and Applied Mathematics*, 61(10):1331–1346, 2008.
- [30] Nick Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Applied and Computational Harmonic Analysis*, 10(3):234–253, May 2001.
- [31] Peter D. Lax. *Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves*. SIAM, 1973.
- [32] Milton C. Lopes Filho, Anna Mazzucato, and H. J. Nussenzweig Lopes. Weak solutions, renormalized solutions and enstrophy defects in 2d turbulence. *Arch. Rational Mech. Anal.*, 179:353–387, 2006.
- [33] Stéphane Mallat. *A wavelet tour of signal processing*. Academic Press, 1999.
- [34] Makoto Matsumoto and Takuji Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation*, 8(1):3–30, 1998.
- [35] Romain Nguyen van yen, Marie Farge, Dmitry Kolomenskiy, Kai Schneider, and Nick Kingsbury. Wavelets meet Burgulence: CVS-filtered Burgers equation. *Physica D*, 237(14–17):2151–2155, 2008.
- [36] Ole Nielsen and Markus Hegland. Parallel performance of fast wavelet transforms. *International Journal of High Speed Computing*, 11:55–73, 2000.
- [37] Paolo Orlandi. *Fluid Flow Phenomena: A Numerical Toolkit*. Springer, 2000.
- [38] Stanley Osher and Fred Solomon. Upwind difference schemes for hyperbolic systems of conservation laws. *Mathematics of Computation*, 38(158):339–374, 1982.
- [39] T. H. Pulliam. Artificial dissipation models for the Euler equations. *AIAA Journal*, 24:1931–1940, December 1986.
- [40] Kai Schneider, Marie Farge, Alexandre Azzalini, and Jörg Züuber. Coherent vortex extraction and simulation of 2d isotropic turbulence. *Journal of Turbulence*, 7:44, 2005.
- [41] Kai Schneider and Oleg Vasilyev. Wavelet methods in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 42, 2010.
- [42] K. R. Sreenivasan. On the scaling of the turbulence energy dissipation rate. *Physics of Fluids*, 27:1048–1051, 1984.
- [43] P. K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 21(5):995–1011, 1984.
- [44] Eitan Tadmor. Convergence of spectral methods for nonlinear conservation laws. *SIAM Journal on Numerical Analysis*, 26(1):30–44, 1989.
- [45] Chuong V. Tran and David G. Dritschel. Vanishing enstrophy dissipation in two-dimensional Navier-Stokes turbulence in the inviscid limit. *Journal of Fluid Mechanics*, 559:107–116, 2006.
- [46] Lloyd N. Trefethen. *Spectral Methods in Matlab*. SIAM, 2000.
- [47] W. Wolibner. Un théorème sur l’existence du mouvement plan d’un fluide parfait, homogène, incompressible, pendant un temps infiniment long. *Mathematische Zeitschrift*, 37(1):698–726, December 1933.
- [48] V. I. Yudovich. On the loss of smoothness of the solutions of the Euler equations and the inherent instability of flows of an ideal fluid. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 10(3):705–719, 2000.